AMYGDALA

$z \mapsto z^2 + c$

A Newsletter of $\mathcal{M}$ -- the Mandelbrot Set
AMYGDALA, Box 219, San Cristobal, NM 87564
$15.00 for ten issues ($25 overseas)

Issue #1a
February 9, 1987

CONTENTS

APOLOGY

I realize with somewhat of a shock that five months have gone by since I sent out the original Amygdala #0; my intention was to get out a newsletter every two months at first, increasing later to once a month. Why the long delay, then? Well, as Voltaire said, "Le mieux est l'ennemi du bien": "The best is the enemy of the good." I've had the desire to put out issue #1 replete with many pictures, color slides and black & whites, but doing the pictures has stretched the process out much longer than I expected. The shock that brought me back to reality came in the form of a letter received yesterday from my oldest friend, who is also a subscriber:

> I hope you are not planning to become a newsletter scoundrel. In the early 70's I was involved in a newsletter called "Pension Fund Report". Our editor was a bright young man from Georgia who had the gift of gab. He called people on the phone, got a lot of information out of them, whipped it into shape and got it printed and mailed. It was good. We were approaching the critical point of renewal time for the initial annual subscriptions, which is also the point at which subscriptions became profitable, when he started falling behind. Further and further behind. Finally stopping. A potentially valuable enterprise became worthless. A couple of large institutions even wanted part of their money back. It was a shame. The man simply became interested in other ventures, which he then abandoned for yet other ventures. Sad but true.

Thanks, Jack; I'm not planning to become a newsletter scoundrel, but I can understand your concern. Herewith, then, subscribers, is issue #1; not the best, perhaps — but I hope you will find it good.

THE AMYGDALAN SECTS

From the Encyclopaedia Galactica, 19th Edition:

"The decline and eventual collapse of Earth's earliest space-faring civilizations had long mystified archaeologists. It was Dr. Martin Dace's work on the Amygdalan Sects which eventually provided an explanation. Dr. Dace was at that time a postgraduate student at the Theological Department of the University of Altair, and had become interested in the psychology of the many bizarre and irrational cults that had sprung up around the time of the collapse. That he picked the Amygdalans for his thesis topic was perhaps pure serendipity. But the skill which he brought to piecing together the strange history of the sect from its fragmentary records shows imaginative genius of the first order.

"Shortly before the collapse, human civilization had constructed its first computers. Crude by our own standards, these machines still used electronic circuitry as a basis for operation. Nevertheless, they were sufficiently powerful to provide glimpses of the **M** Object. The Object, infinitely complex and infinitely beautiful, acted like a lodestone to the finest minds on Earth at that time. They found in it deep mathematical theorems, hinting at insights into phenomena of the natural world; they found aesthetic satisfaction in contemplation of its intricate form; their curiosity led them to explore ever further into its infinite labyrinths. To this task they harnessed the most powerful tools available to them: the computers of industry, of government and of the war lords spent every spare moment focused on the Object.

"The more deeply the Object was investigated, the more alluring it became. Its investigators applied their ingenuity to securing more computer time and access to faster computers. Because the Object's appeal was greatest to the most imaginative and able members of terrestrial society, they were largely successful in gaining access to the resources they required. Openly or surreptitiously, a growing fraction of Earth's intellectual and electronic resources were committed to exploration of the Object. The effects of this were slow to be felt; the diversion of resources was greatest in areas least subject to outside supervision — academia, government laboratories and military research.

"The extent to which the Object had preoccupied Earth's

information-processing systems became dramatically apparent in the 'Fizzle War' between the two major power-blocs on the planet at that time. The political leaders on either side had escalated through a series of ritual insults, provocative gestures and grand-standing to a level from which they could not back down. Having carefully checked the contingency plans for their own survival, they gave orders to commence global thermonuclear war.

"Nothing happened. The computers on both continents were locked in contemplation of the area around *0.0155-0.7i,* where it was suspected that an intriguing new structure lurked. After several embarrassing hours, the politicians found a face-saving formula and backed off.

"This happy outcome, however, presaged a more complete breakdown of society's mechanisms. Computers had penetrated most of the manufacturing and service industries, and as their attention came increasingly to rest upon the Object, the fabric of the industrial civilizations came apart. This collapse fed upon itself, as more and more of those who might have averted disaster chose to turn their attention from the collapsing economy and garbage-strewn cities to the serene beauty of the Object. It was as though the electronic brains that supervised the world's maintenance systems had become absent-minded; flickering traffic signals grid-locked the traffic in city streets; unmonitored reactors melted down or quietly switched themselves off.

"Then the ultimate calamity occurred. The collapsing infrastructure could no longer provide the steady diet of smoothed electrical power that the computers required. One by one, the windows looking in on the Object slammed shut.

"A profound sadness settled over the world. As a philosopher of a slightly earlier period had said, the loss of any pleasing object is painful; the loss of an infinitely pleasing Object must necessarily be infinitely painful. There were, of course, pictures that had been taken before the loss of contact, but these, finite, static, and limited, seemed only to mock the vibrant interaction that had been possible with the Object itself.

"This, then, was the picture Dr. Dace pieced together of the collapse. And this, too, was the origin of the Amygdalan Sects — the small groups that grew out of the collapse, following diverse routes but all motivated by a deep yearning for lost beauty. We append here an extract from Dace's thesis, describing the festival rite of one of the Sects:

"...on the Festival Days, the worshippers gather in the Central Park in the evening. Each, carrying his abacus and the Seven Parasols, goes to his pre-assigned spot; the priests assist any who are in difficulty. At sunset, the High Priest reads out the Objective Coordinates of the north-east and south-west corners of the park; his words are passed through the assembly by special messengers, and the congregation then sits down to compute. Each first computes his own Objective Coordinates, and then Iterates, intoning the the mantra of the sect — the MO mantra. On reaching a result of magnitude greater than twice Unity, he sets aside his abacus and sits in silent meditation. From time to time the priests pass among the congregation, checking the total iterations that each of the meditators has achieved.

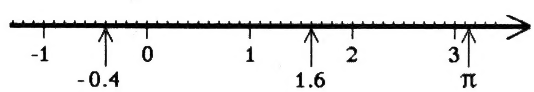
"Just before dawn on the following day, all cease their iterations and wait. From the Tower of Objective Vision, the High Priest announces the numbers limiting the seven ranges. Then, just as the sun rises, each worshipper erects the parasol corresponding to his calculated number. The effect is incomparably beautiful , though it can only be appreciated fully from the top of the Tower..."

—*John Dewey Jones*

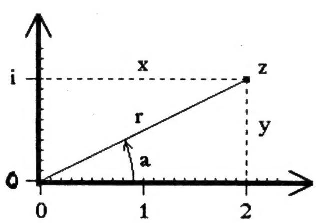
COMPLEX NUMBERS: THE BASICS

Some of you don't have the math necessary for a proper understanding of how Julia sets and the Mandelbrot set are generated. I'm writing this brief introduction to the basics for those who need & want it.

I'll assume that you are all familiar with the real numbers, and their geometric interpretation as points on a "number line":



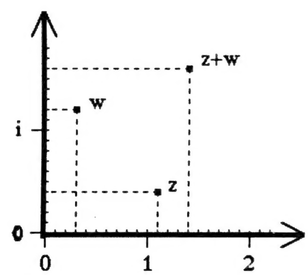
Geometrically, the **complex numbers** correspond to points in a plane -- the **complex plane**. Each point in the plane corresponds to a complex number, and vice versa. In the following picture, there are two coordinate axes: the horizontal one is the **real** axis and the vertical one is the **imaginary** axis. Arrayed along the horizontal axis are the ordinary real numbers: 1,2,3 and the rest, and all the numbers in between. They are a special case of complex numbers. Arrayed along the vertical axis are the **imaginary** numbers -i, -2i, etc.



Each point/complex number has four properties: a **real part**, an **imaginary part**, a **modulus**, and an **argument**. For the point **z** shown in the picture, the real part is **x** = 2, the imaginary part is **y** = 1, the modulus is **r** = √3, and the argument is **a** = arctan 1/2 = 26.57°.

The complex number **z** can be expressed in terms of its real and imaginary parts: z = x + iy = 2 + i.

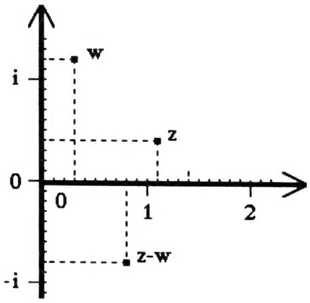
To add two complex numbers, we merely add separately their real and imaginary parts. Thus if z = 1.1 + .4i and w = .3 + 1.2i, then z + w = (1.1+.3) + (.4+1.2)i = 1.4 + 1.6i.



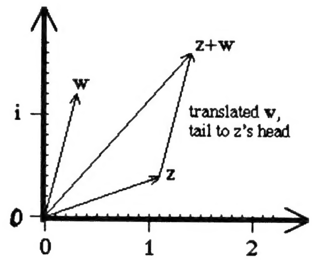
To subtract two complex numbers, we merely subtract separately their real and imagin-

ary parts. Thus for the z and w given above, z - w = (1.1-.3) + (.4-1.2)i = .8-.8i.

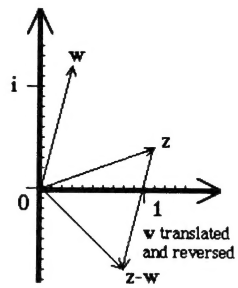
Addition and subtraction can also be performed using arrows pointing from the origin (point 0) to the points in question.



To add w to z, translate the arrow for w (move it without changing either its length or direction) until its tail coincides with z's head. The sum z+w corresponds to the arrow from the origin to the head of the translated arrow.

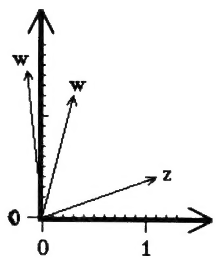


To subtract w from z, reverse the arrow for w, head for tail, and translate it until its (new) tail coincides with z's head. The difference z-w corresponds to the arrow from the origin to the head of the translated arrow.



Multiplying complex numbers is simple! You just multiply their moduli and add their arguments to get the modulus and argument of the product.

Since i has modulus 1 and argument 90°, i^2 = i·i has modulus 1 and argument 180°: -1 in other words. So i is a square root of -1! (So is -i).



We can multiply two numbers z = x+iy and w = u+iv in terms of their coordinates; we can group (associate) operations, interchange arguments, and distribute addition through multiplication just as for the real numbers; therefore: zw = (x+iy)(u+iv) = xu + xiv + iyu + iyiv. Doing a bit of juggling, and realizing that iyiv = iiyv = i^2yv = -yv, we get: zw = (xu-yv) + i(xv+yu) and that's usually the best way to multiply two complex numbers. For the particular example we've been using, zw = (1.1+.4i)(.3+1.2i) = (1.1x.3-.4x1.2) + (1.1x1.2+.4x.3)i = -.15 + 1.44i.

(to be continued)

SPEEDUP? AN EXCHANGE

Several readers have questioned the usefulness of the tests or "tricks" used in my Macintosh program, described in the last issue under "SPEEDUP". The purpose of the three tests is to help determine if a point is in $\mathcal{M}$ without calculating all the iterates up to the dwell limit. Please recall that the **iterates** of a complex number c are the numbers $z_0 = 0$, $z_1 = z_0^2+c$, $z_2 = z_1^2+c$, ... The **dwell** of c is the smallest n for which $|z_n| > 2$. If all $|z_n| \le 2$, i.e. $c \in \mathcal{M}$, then the dwell is infinite.

The code for the tests increases the time to calculate each iterate, but this increase is offset by a decrease in the number of iterates that need to be calculated, at least for some points in $\mathcal{M}$. In a region poor in points belonging to $\mathcal{M}$ code for the tests will actually increase the computation time.

The compute cost of the tests should not be exaggerated, however. The "slow/fast trick" described in Dewdney's column in Scientific American, is remarkable because it can detect a repetition in an arbitrarily long series using only two variables. The price paid is the extra cost of computing the (redundant) "slow" value -- a 50% increase in the number of multiplications, in our case. I used the idea of the slow/fast trick without incurring any extra multiplications by retaining an array of computed values from which the "slow" value is retrieved instead of being recalculated.

As the successive iteration values $z_0, z_1, z_2, ...$ are calculated, they are stored in an array as Z[0], Z[1], Z[2],... Each test then consists of comparing the currently computed value z_n (n > 0) with some stored value Z[k].

The loop test, corresponding to the slow/fast trick, compares z_n with Z[n/2]. The convergence test compares z_n with Z[n-1], to see if two successive values are the same. The cycle test compares z_n with Z[0].

The listing of an assembler program for computing dwell is given later on in this issue, together with a detailed discussion. This program originally included all three tests. When applied to the calculation of $\mathcal{M}$ itself, the loop test accounted for 62% of the points in $\mathcal{M}$, while the convergence test accounted for 29%, and the cycle test for less than 1%. Thus it would seem that the cycle test is unproductive, and it has been dropped.

An exchange between Mark Bolme and RS about the pros & cons of these tests follows.

Mark Bolme writes:

I was interested in the "tricks" used in your Mac program. However, I question their utility. Their advantages are a reduced number of iterations for points whose dwell is above the dwell limit. Their disadvantage is that they add a small overhead to EACH iteration for EVERY point.

It seems that their utility is based on the following:

1) The fraction of points whose dwell exceeds the dwell limit (D);

2) The average reduction in the number of iterations for points whose dwell exceeds D;

3) The average number of iterations for points whose dwell does not exceed D;

4) The increase in iteration time needed to implement the "trick".

Discussing each of these points in turn:

1) Our experience [at Sintar] has been that people tend to look at pictures with many more colored points than black points. There are exceptions, but they are rare. Calculation over the entire Mandelbrot Set will exaggerate the effect of the "tricks", because of the large amount of black (compare the amount of black in the picture on page 17 of the **Scientific American** article with the black in the remaining pictures). Of course this may depend on the system used. People with black and white displays may look at more "black", because there is no color to display.

2) I do not have any data here. Perhaps you do.

3) Our experience has been that people tend to look at pictures with high iteration counts. Here, as in point (1), calculation over the entire Mandelbrot Set will exaggerate the effect of the "tricks", because the surrounding regions have very low dwell (average 8.4 by your estimation).

4) I do not have any data here, either. Perhaps you do. Clearly, there is a break even point. I have no doubt that when calculating over the whole Mandelbrot set it is best to use all possible tricks, but how about for the "average picture"?

The point that I am leading up to is that there needs to be some sort of fair benchmark for Mandelbrot calculating routines/programs. If you come up with one, I will be glad to comply. When considering a fair benchmark, the points mentioned above should be considered, but also one needs to recognize the different array sizes and screen aspects for different computers.

In the absence of a benchmark, I propose calculating -0.744353 + 0.113454i at a magnification of 200,000. If I haven't made any typos, that should give everyone a screen filled with points of dwell 50. Then divide the time in seconds to calculate by the total number of points calculated. I have run this benchmark for **FractalMagic**, and the results are:

IBM PC Clone at 8 MHz
- with 8087 chip: 94.0 points per second.
- w/o 8087 chip: 6.24 points per second.

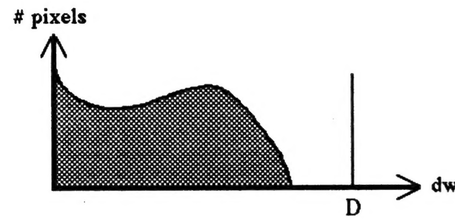
Finally, your readers might be interested in the details of our programs. We support Apple and IBM. Our Apple programs support Double Hi-Res, our IBM programs support EGA and the 8087 chip. All our programs allow you to select the region to calculate directly from plots, and also allow you to stop the calculations, save them and resume them later. I am including some further information regarding our programs, and would be glad to submit them for review.

Comment by RS:

I think Mark is correct in his criticism of the "tricks". For color pictures of $\mathcal{M}$ without a good deal of black the game may not be worth the candle; the grand view of $\mathcal{M}$ is an exceptional case.

I should point out one type of exploration in which the "tricks" may prove worthwhile, however, and that is in determining what dwell limit to use for a given picture. ("Dwell limit": We are speaking of course of the limit D to the number of iterations z R z^2+c a program performs on a point without exceeding the "explosion limit" $|z| > 2$ before it gives up and assumes that c is in $\mathcal{M}$.)

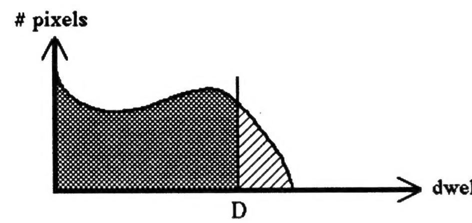
The general effect of an insufficiently large dwell limit is, I believe, to broaden and coarsen the picture, obscuring the fine detail (I think this is the case, for example, in ArtMatrix's fractal image CB2B4A5C/24 -- care to check this out, Homer Wilson Smith?), as opposed to mere blurring, which is due, I think, to insufficient precision in the underlying arithmetic. How, then, is one to determine what is "sufficiently large"? The only ways I can think of are (1) to try larger and larger D, until the picture does not change appreciably, or (2) to try a **huge** D, expecting (at a particular



grain size) to get a distribution of dwells looking something like this:

the pileup at D consisting of those pixels corresponding to points actually in $\mathcal{M}$.

If the pile is nonempty, then as D grows larger, the time lost in applying the "tricks" to the pixels not in $\mathcal{M}$ (the grey area), will eventually be more than made up by the savings



the tricks provide for the pixels in $\mathcal{M}$.

If D is chosen too small, the distribution will look like this:

the pixels in the hatched region being erroneously attributed to $\mathcal{M}$, hence the broadening and coarsening: those pixels will appear black and be merged into $\mathcal{M}$ (or a nearby replica thereof) rather than being distinguished from it by color.

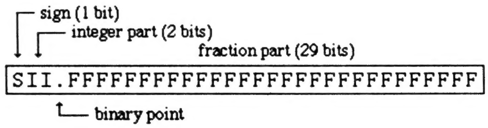
Is anyone not connected with Sintar interested in reviewing their programs for Amygdala? Unfortunately I can't, having only a Mac and access to a Sun 3. To inquire about Sintar's products:

DWELL: AN ASSEMBLER PROGRAM

The listing in the facing column is a program written for the 68000 microprocessor in Assembler, which computes the dwell for a complex number c = u+iv, using dwell cutoff limit D.

The program is incomplete, lacking a preamble, which saves working registers as appropriate, places u in 68000 register D0, v in D1, and D in D2. It also lacks a postscript, which stores the result appropriately and restores the working registers.

u and v are expressed in a special 32-bit fixed-point format, which I call **B29** (29 fraction bits):



negative values are represented as the two's complement value of the corresponding positive value, as usual. For example, the hex values for +1 and -1/2 are:

```
+1:     20000000
-1/2:   F0000000
```

The code fragment leaves the iteration count in register D2 and a result code in D0: zero means c is outside $\mathcal{M}$, nonzero means c is in $\mathcal{M}$, or an error occurred: 1 means the dwell limit was achieved; 2 means the loop test succeeded; 3 means the convergence test succeeded; 4 means the specified dwell argument D exceeded the maximum allowable, DWELLM, defined arbitrarily to be 1024; change it as you see fit.

After the line defining DWELLM, the next three lines of code specify three data structures private to this dwell routine: the HISTORY array, containing what we have called Z[n] above; NMAX, which holds the dwell limit argument, and SIGN, which is used in the double-precision multiplication. We assume that these three lines specify offsets relative to some data space base address, rather than absolute addresses. For the Macintosh, the data space base address is held in register A5.

The 68000 registers D0-D7 and A0-A3 are assigned as follows:

D0	u,x	Real part of c, z.
D1	v,y	Imaginary part of c, z.
D2	d,n	Specified dwell cutoff limit, iteration index.
D3	xsq	x squared.
D4	ysq	y squared.
D5	a	
D6	b	
D7	t	Temporary.
A0	half	Points at Z[n/2].
A1	last	Points at Z[n].
A2	u	
A3	v	

[1] The program proper STARTs with a comparison to insure that the dwell cutoff limit D supplied in D2 does not exceed DWELLM, else an error indication is returned.

[2] The pointer to **half**, which is held in A0, is initialized pointing to the first double longword of the HISTORY array, as is the pointer to **last**, held in A1.

[3] u and v, held in A2 and A3, are initialized from D0 and D1, as supplied initially. The values left in D0 and D1 are the initial values of x and y, where z = x+iy.

[4] D is moved from D2, as supplied, to NMAX; then the iteration index n, held in D2, is initialized to zero.

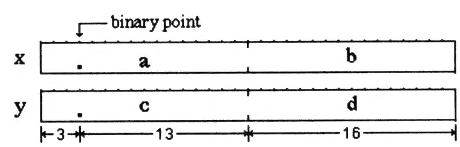
[5] The iteration loop starting at NEXT, begins by storing z = x+iy into Z[n], as pointed to by **last**.

[6] The next 7 lines set x and y equal to their absolute values, preparatory to multiplying them together, and compute and save the sign of the result.

```
DWELLM   EQU      1024

HISTORY  DS.L     2*DWELLM
NMAX     DS.L     2
SIGN     DS.L     1

START:   CMP.L    #DWELLM,D2         ; [1]
         BGT      ERROR
         LEA      HISTORY(A5),A0     ; [2]
         MOVE.L   A0,A1
         MOVE.L   D0,A2              ; [3]
         MOVE.L   D1,A3
         MOVE.L   D2,NMAX(A5)        ; [4]
         CLR.L    D2
NEXT:    MOVE.L   D0,(A1)            ; [5]
         MOVE.L   D1,4(A1)
         MOVE.L   D0,SIGN(A5)        ; [6]
         BGE.S    L1
         NEG.L    D0
L1:      TST.L    D1
         BGE.S    L2
         NEG.L    D1
         NEG.L    SIGN(A5)
```

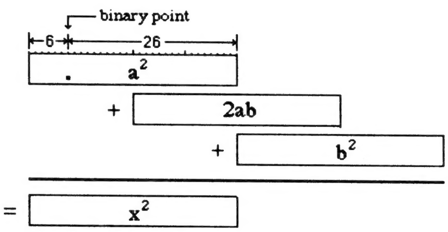


We have |x| = a+ib and |y| = c+id:

We carry out one step of the iteration by computing $z^2+c = (x+iy)^2 + (u+iv) = (x^2-y^2+u) + i(2xy+v)$. We calculate the squares and products rapidly by making use of the unsigned multiply instruction (MULU) built into the 68000. Complications arise from the fact that (1) MULU multiplies two 16-bit quantities together, which we must use to build 32-bit

multiplies; and (2) multiplying two format B29 32-bit fractions of the form `0II.FF...F` together produces a 64-bit fraction of the form `0IIIII.FF...F`, of which we retain the left 32 bits. The main point to be noted is that the binary point is **six** places to the right in a product or a square (format B26), not three.

Letting $R = 2^{16}$, $B = 2^3$, a = the unsigned integer represented by the left 16 bits of $|x|$ and b represented by the right 16 bits, we have $|x| = B(a+b/R)/R$ and $|y| = B(c+d/R)/R$. We then have $x^2 = B^2(a^2 + 2ab/R + b/R^2)/R^2$. Similarly $y^2 = B^2(c^2 + 2cd/R + d/R^2)/R^2$, and $|xy| = B^2(ac + (ad+bc)/R + bd/R^2)/R^2$.



We compute x^2 as the sum of three partial products:

[7] Compute b^2 in D3, first as a 32-bit product. The next three instructions round it into its high order 16 bits, leaving Bb^2/R in D3.

[8] Move and swap a and b into D5 lo and D7 lo, respectively, then multiply them together to form ab in D7. Add D7 to itself to form 2ab, then add it to D3 to form $B(2ab+b^2/R)$ there.

[9] Round and swap D3 to form $B(2ab/R+b^2/R^2)$.

[10] Form a^2 in D7, then add D3 to it to form $x^2 = B(a^2+2ab/R+b^2/R^2)$ in D7, scaled B26.

[11] Form y^2 similarly in D4.

[12] Form x^2+y^2 in D7 and exit to OUT if it is larger than 4 (note the B26 scaling).

[13] Form 2xy in D2, scaled B29.

Recall that at this point u is in A2, v is in A3, x^2 is in D3, y^2 is in D4, and 2xy is in D1.

[14] Form y = 2xy+v for the next iteration in D1; note that 2xy is already scaled B29.

[15] Form $x = x^2-y^2+u$ for the next iteration in D0, scaling

```
L2:     MOVE.L  D0,D3           ; [7]
        MULU    D0,D3
        ADD.L   #$8000,D3
        CLR.W   D3
        SWAP    D3
        MOVE.L  D0,D5           ; [8]
        SWAP    D5
        MOVE.L  D0,D7
        MULU    D5,D7
        ADD.L   D7,D7
        ADD.L   D7,D3
        ADD.L   #$8000,D3       ; [9]
        CLR.W   D3
        SWAP    D3
        MOVE.L  D5,D7           ; [10]
        MULU    D7,D7
        ADD.L   D7,D3
        MOVE.L  D1,D4           ; [11]
        MULU    D1,D4
        ADD.L   #$8000,D4
        CLR.W   D4
        SWAP    D4
        MOVE.L  D1,D6
        SWAP    D6
        MOVE.L  D1,D7
        MULU    D6,D7
        ADD.L   D7,D7
        ADD.L   D7,D4
        ADD.L   #$8000,D4
        CLR.W   D4
        SWAP    D4
        MOVE.L  D6,D7
        MULU    D7,D7
        ADD.L   D7,D4
        MOVE.L  D3,D7           ; [12]
        ADD.L   D4,D7
        CMP.L   #$10000000,D7
        BGT.S   OUT
        MOVE.L  D0,D7           ; [13]
        MULU    D1,D7
        ADD.L   #$8000,D7
        CLR.W   D7
        SWAP    D7
        MULU    D6,D0
        ADD.L   D7,D0
        MULU    D5,D1
        ADD.L   D0,D1
        ADD.L   #$8000,D1
        CLR.W   D1
        SWAP    D1
        MULU    D6,D5
        ADD.L   D5,D1
        TST.L   SIGN(A5)
        BGE.S   L3
        NEG.L   D1
L3:     ASL.L   #4,D1
        ADD.L   A3,D1           ; [14]
        MOVE.L  D3,D0           ; [15]
        SUB.L   D4,D0
        ASL.L   #3,D0
        ADD.L   A2,D0
        ADDQ.L  #1,D2           ; [16]
```

x^2-y^2 B29 before adding u.
[16] Increment the interation index n in D2.
[17] Compare z = x+iy with Z[n/2], i.e. D0 with what half = A0 addresses and D1 with the following long word, to see if we have looped.
[18] Compare z = x+iy with Z[n/2], i.e. D0 with what last = A1 addresses and D1 with the following long word, to see if we have converged.
[19] Compare n with NMAX to see if we have reached the iteration limit, whence c is in $\mathcal{M}$.
[20] Increment **last** and, if n is odd, increment **half**. Loop back to NEXT for the next iteration.
[21] The number of iterations performed so far is in D2;

```
         CMP.L    (A0),D0         ;[17]
         BNE.S    L4
         CMP.L    4(A0),D1
         BEQ.S    LOOPED
L4:      CMP.L    (A1),D0         ;[18]
         BNE.S    L5
         CMP.L    4(A1),D1
         BEQ.S    CONVRG
L5:      CMP.L    NMAX,D2         ;[19]
         BGE.S    IN
         ADDQ.L   #8,A1           ;[20]
         BTST     #0,D2
         BEQ      NEXT
         ADDQ.L   #8,A0
         BRA      NEXT

IN:      MOVEQ.L  #1,D0           ;[21]
         BRA.S    EXIT
LOOPED:  MOVEQ.L  #2,D0
         BRA.S    EXIT
CONVRG:  MOVEQ.L  #3,D0
         BRA.S    EXIT
ERROR:   MOVEQ.L  #4,D0
         BRA.S    EXIT
OUT:     MOVEQ.L  #0,D0
EXIT:    ...
```

place the appropriate result code in D0, and go to EXIT to perform the postscript, deal with the iteration number and result code appropriately, and restore the working registers.

REVIEW: "ON GROWTH AND FORM"

On Growth and Form: Fractal and Non-Fractal Patterns in Physics.
Edited by H. Eugene Stanley and Nicole Ostrowsky.
Martinus Nijhoff Publishers, 1986; 308 pages, paperback.
Reviewed by Thomas Bank.

On Growth and Form is the culmination of a series of lectures and seminars held from 26 June to 6 July 1985 at the Institut d'Etudes Scientifiques de Cargese in Corsica, France. This event featured speakers and students from around the world sharing their work and interests in fractals. The book is divided into two major sections. The eleven chapters of Section A introduce fundamental principles of form, growth, and fractals. These tutorial chapters are based on the opening lectures presented by the speakers. Topics include self-similarity and fractal behavior, scale-invariant growth, computer simulation of growth, and the diffusion limited aggregation growth model (DLA).

The remaining twenty-two chapters that make up Section B are technical descriptions of the work and results of researchers working with fractals. The topics presented here include Monte Carlo techniques, crack propagation and failure, the dynamics of fractals, and flow through porous materials. Mathematically inclined readers will enjoy the rigor with which each subject is covered. Each chapter is concluded with an excellent bibliography pointing the reader to related references.

While **On Growth and Form** does not discuss the Mandelbrot set in particular, readers with an interest in fractals will find the many ideas in this book stimulating and

BIBLIOGRAPHY

Items #1-7 appeared in Amygdala #0 previously. #8-24 were submitted by Allen B Freedman.

1. AK Dewdney, Computer Recreations column, Scientific American, August 1985, cover & pp 16ff. [Most of us "discovered" the Mandelbrot set by reading this piece. Lots of useful information, plus stunning pictures by Peitgen & Richter.

2. BB Mandelbrot. The Fractal Geometry of Nature. WH Freeman & Co, New York (1983) ISBN 0-7167-1186-9. [The basic work on fractals by the pioneer, Mandelbrot. The book is a feast for the eyes and the mind, and a "must" for anyone who wants to dig deeper into the world of fractals.]

3. HG Schuster. Deterministic Chaos. Physic-Verlag (1984) (220 pages plus plates). ISBN 0-89573-223-8, VCH Publishers, 303 NW 12th Avenue, Deerfield Beach, FL 22441-1705. [This is a technical book about physical systems which behave chaotically, i.e. with disorder and irregularity. There seems to be a deep connection between the dynamics of such systems and the process which underlies $\mathcal{M}$. Plates IX through XV are georgeous renditions of $\mathcal{M}$-views, taken from Peitgen and Richter (1984; see #4).]

4. H-O Peitgen & PH Richter. Harmonie in Chaos und Kosmos; and Morphologie komplexer Grenzen; Bilder aus der Theorie dynamischer Systeme. Forschungsschwerpunkt: Dynamische Systeme, Universität Bremen, D-2800 Bremen, F.R.G.

5. H-O Peitgen & PH Richter. The Beauty of Fractals. Springer-Verlag New York (1985). ISBN 0-387-15851-0: 110 pages & 184 figures in 221 separate illustrations, mostly in color. [If you have only one book about fractals and

$\mathcal{M}$, this should be it! Beggars description. The following is quoted from the publisher's blurb about the book: "This book represents an unusual attempt to publicize the field of Complex Dynamics. The editors report on this rapidly developing discipline in terms of computer graphical pictures that resulted from their own studies of iterated maps. Such maps arise, e.g. in the problem of root finding, or in the renormalization group theory of phase transitions. They define highly complex boundaries between various domains of attraction, also known as Julia sets for rational maps of the complex plane. This volume contains over 70 full color pictures of Julia sets, including some that represent Yang-Lee zeroes for hierarchical models of magnetism, of corresponding Mandelbrot sets analyzed to high resolution, and of the domain structure of real analytic maps.

"Contributions by noted experts complement the book. B.B. Mandelbrot gives a very personal account of his discovery of the Mandelbrot set, and on the history of fractals in general. H.W. Franke, a pioneer in computer graphics, comments on the relationship between science and art. G. Eilenberger, as a theoretical physicist, points to the changes that our understanding of natural phenomena may undergo in the study of nonlinear processes. Finally A. Douady, one of the leading experts in the mathematics of the Mandelbrot set, recounts what is known about this amazingly complex object and what is not.

"**The Beauty of Fractals** is not meant to be a textbook, nor is it just a collection of attractive computer graphics. It is an experiment in that it presents mathematical concepts in a form which will appeal to everyone from layman to expert: it provides the latter with the material to reflect on, while the former is invited to marvel at the beauty of mathematics."]

6. A Douady & JH Hubbard. "Itération des polynômes quadratiques complexes". C.R. Acad. Sci. Paris Sér. I Math. 294 (1982), no.3, 123-126. ["Let **M** be the set of points $c \in \mathbf{C}$ for which the Julia set of $z \to z^2+c$ is connected. We show that **M** itself is connected, and a description of **M** is given." (from the abstract)]

7. L Dreiling. "Fractal Art". Computer Graphics World (July 1986) 91-92. [A short introduction with four fairly nice color pictures]

8. M Davis. Interview with Benoit B. Mandelbrot. Omni 6 (February 1984) 65-66, 102-107.

9. C Dodge & CR Bahn. "Musical Fractals" Byte 11 (June 1986) 185-196. [Includes programs "written in MSX BASIC extended with the MUSIC MACRO commands for the Yamaha CX5-M music computer."]

10. KJ Falconer. The Geometry of Fractal Sets. Cambridge University Press, New York, 1985. Cambridge Tracts in Mathematics #85. 162 pp., illus., $32.50. [see Taylor's review (#23), below]

11. J Gleick. "The Man Who Reshaped Geometry". The New York Times Magazine, 8 December 1985, 64-65+.

12. LP Kadanoff. "Fractals: Where's the Physics?". Physics Today 39 (February 1986) 6-7.

13. BR Land. "Dragon". Byte 11 (April 1986) 137-138. [Dragon curves drawn with MacFORTH]

14. BG Levi. "New Global Fractal Formalism Describes Path to Turbulence". Physics Today 39 (April 1986) 17-18. [Rayleigh-Bénard experiment; work of Thomas C. Hardy et al]

15. M Lewis & DC Rees. "Fractal Surfaces of Proteins". Science 230 (6 December 1985) 1163-1165.

16. S Lovejoy. "Area-Perimeter Relation for Rain and Cloud Areas". Science 216 (9 April 1982) 185-187.

17. J McDermott. "Geometrical Forms Known as Fractals Find Sense in Chaos". Smithsonian 14 (December 1983) 110-117.

18. A Norton. "Geometric Fractals in 3-D". SIGGRAPH, July 1982.

19. R Orbach. "Dynamics of Fractal Networks". Science 231 (21 February 1986) 814-819.

20. D Pountain. "Turbocharging Mandelbrot". Byte 11 (September 1986) 359-366.

21. AL Robinson. "Fractal Fingers in Viscous Fluids". Science 228 (31 May 1985) 1077-80.

22. PR Sørenson. "Fractals". Byte 9 (September 1984) 157-172. [Includes Applesoft BASIC program]

23. SJ Taylor. Review of The Geometry of Fractal Sets, by KJ Falconer. Science 229 (27 Sept 1985) 1381.

24. M van de Panne. "3-D Fractals". Creative Computing 11 (July 1985) 78-82. [Includes BASIC programs for TRS-80 Models I/II/III/4 (with CGP-115 plotter), Macintosh, and Apple II].

THE BEAUTY OF FRACTALS lists for $35.00. Amygdala subscribers can get it from us for $31.50, postpaid. Please allow 4-6 weeks for delivery, ARO.

COMMERCIAL PRODUCTS

ART MATRIX; PO Box 880; Ithaca, NY 14851 USA. (607) 277-0959. Prints, FORTRAN program listings, 36 postcards $7.00, sets of 2 packs $10.00, 140 slides $20.00. Or send for FREE information pack with sample postcard. Custom programming and photography by request. Make a bid.